UOT: 658.012.2

MODEL ƏSASLI KOD GENERASİYASINDA YENİ YANAŞMANIN (EDA YANAŞMASININ - 2017) QİYMƏTLƏNDİRİLMƏSİ

N. Ş. SƏDİLİ, C. N. İSMAYILZADƏ
Bakı Mühəndislik Universiteti

Proqram mühəndisliyindəki son məsələlərdən biri proqram təminatı inkişafı prosesinin avtomatlaşdırılmasıdır. Proqram təminatı inkişafını səmərəli şəkildə avtomatlaşdırmaq, inkişaf prosesində insan əməyini azaltmaq və sürəti artırmaq üçün müxtəlif cəhdlər edilir. Bu məqalə proqram təminatı inkişafının avtomatlaşdırılması ilə bağlı aktual olan problemlərdən birini – model əsaslı kod generasiyasını təqdim edir və yeni EDA yanaşmasını izah edir. Bu təsvir xarakterli bir məqalədir və iş əsnasında keyfiyyət təhlili metodu istifadə edilmişdir.

Açar sözlər: Proqram mühəndisliyi, proqram təminatı inkişafı həyat dövrü, model-görünüş-müfəttiş arxitekturası.

Proqram mühəndisliyində (PM) (SE-Software Engineering) ən önəmli və eyni zamanda ən diqqətəlayiq sahələrdən biri proqram təminatının inkişafı (PTİ)(SD- Software Development) hesab olunur. PTİ-ın yoxlanma və nəzarəti prosesini daha asan etmək üçün çox sayda fərqli layidə idarəetməsi modelləri və ya başqa sözlə proqram təminatı inkişafı həyat dövrü (PTİHD) (SDLC- Software Development Life Cycle) modelləri işlənilib hazırlanmışdır. Bunlardan bəzilərinə misal olaraq xətti(linear) və ya şəlalə(waterfall) modelini, spiral modelini, çevik (agile) modeli [1] və başqalarını misal göstərmək olar.

PTİHD-ün ən önəmli mərhələlərindən biri kodlaşdırma mərhələsidir. Sistem analizi (SA) və kodlaşdırma, bəzi mənbələrə görə isə proqram təminatı inkişafı və ya yazılımı (development) ən kritik mərhələlərdəndir, belə ki, sistem analizi mərhələsindəki hər hansısa yalnışlıq, son məhsulda – proqram təminatında problemlərə gətirib çıxara bilər, çünki əgər biznes tələbləri doğru analiz edilməzsə son məhsul müstərinin istədiyinin eynisi olmayacaqdır. Bununla yanaşı analiz yaxşı olsa belə, əgər kodlaşdırmada yalnışlıqlar olsa bu halda da son məhsul lazım olduğu kimi çalışmayacaq.

Məqalədə istifadə edilmiş araşdırma metodu keyfiyyət təhlili metodudur. Məqalə üçün istifadə edilmiş olan materiallar müxtəlif məqalələrdən və 2016-cı ilin yanvar ayından bəri üzərində çalışdığımız bir layihədən götürülmüşdür.

Model əsaslı kod generasiyasında yeni yanaşmanın (EDA yanaşmasının - 2017) qiymətləndirilməsi.

Biz ancaq kodlaşdırma mərhələsinin üzərində duracağıq. Daha öncə də qeyd edildiyi kimi, kodlaşdırma PTİHD-ün ən kritik mərhələlərindən biridir, o çox vaxt aparır, dolayısıyla da çox xərc tələb edir. Bu vaxt və xərci azaltmaq üçün müxtəlif kod generasiyası texnikaları işlənib hazırlanır. Bu modellərə nəzər saldıqda, onların böyük əksəriyyətində istifadə olunan yanaşmalarda ortaq cəhətin kodlaşdırmada insan əməyini minimuma endirmək olduğunu görə bilirik. Son trendlər insan əməyini tam sıfıra endirməyə çalışır, daha dəqiq desək kodlaşdırmanı sıfıra endirərək proqramçı mühəndis olmayan insanlar tərəfindən də PTİ-nı reallaşdırmağa çalışır. Bu sahədəki cəhdlərdən bir neçəsinə misal olaraq, avtomatlaşdırılmış kod generasiyası modellərini (AKG) (ACG-Automated Code Generation), AKG üçün komputer dəstəkli proqram mühəndisliyi (KDPM) (CASE-Computer-Aided Software Engineering) alətləri, AKG üçün vahid inkişaf mühitləri (VİM) (İDE-İntegrated Development Environment) [2] və başqalarını göstərmək olar. Ancaq qeyd etmək lazımdır ki, bu sahədə həddən artıq kiçik və ya böyük miqyasda istifadə olunan texnikalar və alətlər mövcuddur.

Öncə model-görünüş-nəzarətçi (MGN)(MVC-Model View Controller) arxitekturasını səthi nəzərdən keçirək. MGN ən məhşur və çox istiadə olunan bir arxitektura olub, bütöv proqram təminatını müxtəlif parçalara, daha dəqiq desək model, görünüş və nəzarətçi olmaqla üç tam müstəqil hissəyə ayırır. Görünüş hissəsi, modelin vəziyyətini, ondan gələn məlumatların təqdim edilməsindən məsul olan hissədir. Nəzarətçi hissəsi görünüş hissəsindən istifadəçi tərəfindən daxil edilmiş hissənin yoxlanması və daha sonra işlədilməsi üçün model hissəsinə göndərilməsi ilə məşğul olur. Verilənlərin və ya məlumatların işlənməsi baxımından MGN arxitekturasının ən əsas hissəsi model hissəsidir, görünüş və nəzarət hissələrindən keçən məlumat məntiqi əməliyyatlar üçün buraya ötürülür [3]. Bu məqalənin əsas məqsədi model generasiyasını optimallaşdırılması üçün yeni yanaşma təklif etməkdir.

MGN arxitekturasını istifadə edən bir çox freymvörklərdə bir sinif sırf model hissəsi üçün

ayrılır və məlumatların alınması, işlənməsi və ötürülməsi kimi məntiqi əməliyyatlar hamısı və ya çoxu bu sinifdə həyata keçirilir. Nəticə etibarı ilə bu sinif çox qarışıq və struktursuz olmuş olur, çünki sorğudan asılı olaraq xeyli fərqli sayda mümkün ehtimallar ola bilər. Kodun daha çox xaotik olması, generasiyasının daha çətin olmasına gətirib çıxarır. Bu cür çətinliklərdən yayınmaq üçün biz xarici servis-məlumat-algoritm (XMA)(EDA- External-Data-Algoritm) modelini təklif edirik. Bu MGN arxitekturası modeli üçün yeni bir yanaşmadır.

XMA modeli MGN modelində tələb oluna biləcək bütün mümkün əməliyyatları üç kateqoriyaya bölür: verilənlər bazası obyektləri ilə əlaqəli əməliyyatlar, əsasən CRUD əməliyyatları, verilənlər bazası ilə, oxuma, yazma, dəyişmə və silmə əməliyyatları, verilənlərin işlənməsi ilə əlaqəli olan əməliyyatlar – mürəkkəb alqoritmlər və s. və qalanlar, hansılar ki, verilənlərin ötürülməsi və qəbul edilməsi ilə əlaqədardırlar. Şəkil 1-də XMA modeli bütöv MGN arxitekturası içərisində təsvir edilmişdir.Şəkildən də göründüyü kimi, ənənəvi modellərdən fərqli olaraq, XMA modeli fərqli kateqoriyalara məxsus əməliyyatları bir birindən ayıran üç blokdan ibarətdir.

Məlumat bloku (Data block) modelin ən önəmli blokudur, o verilənlər bazası ilə əlaqəli əməliyyatlardan məsuldur. Daha öncə qeyd edildiyi kimi, CRUD əməliyyatları da burada həyata keçirilir. Əgər sorğu sadəcə CRUD əməliyyatları ilə bağlıdırsa, o ümumi model tərəfindən qəbul olunur, birbaşa bu bloka ötürülür və bu blok işlənilir.

Ancaq heç də bütün sorğular tək CRUD əməliyyatları ilə bağlı olmur, adətən istifadəçi tərəfindən girilən və ya verilənlər bazasından çəkilmiş olan məlumatların üzərində bəzi məntiqi əməliyyatlar da tələb olunur. Sırf bu səbəbdəndir ki, bir modelə müstəqil **algoritm bloku** da əlavə etmişik. O data bloku ilə yanaşı işləyərək, bu tip əməliyyatları həyata keçirir.

məlumat və xarici servis bloklarının müstəqil və ya yanaşı işləyərək müxtəlif tipli əməliyyatları həyata keçirməsini izah etdik. Ancaq bütün tip əməliyyatlar bu iki blok ilə kifayətlənmir. Bunlarla yanaşı bəzi əməliyatlar var ki, başqa informasiya sistemləri ilə məlumat alış-verişi ilə məşğul olur, məs. bir çox bank əməliyyatları bu tip əməliyyatlardan ibarət olur. Günümüzdə servis yönümlü proqramlaşdırma populyar olduğundan dolayı bu tip əməliyyatlara daha çox rast gəlinir. Bu əməliyyatlar bir informasiya sistemi daxilində və ya başqaları ilə arasında həyata keçirilə bilər.

Bu əməliyyatların kateqoriyalarını nəzərə alsaq, onların ardıcıllıqları model generasiyası zamanı çox önəm daşıyır. Fərqli sorğu tipləri əsasında müxtəlif ardıcıllıqlardan hazırlanmış önəmli permutasiyalar aşağıda verilmişdir:

- Məlumat bloku
- Algoritm bloku
- Algoritm bloku -> Məlumat bloku
- Məlumat bloku -> Algoritm bloku
- Məlumat bloku -> Algoritm bloku -> Xarici servis bloku
- Xarici servis bloku -> Algoritm bloku -> Məlumat bloku
- Xarici servis bloku -> Məlumat bloku
- Məlumat bloku -> Xarici servis bloku

Nəticə

Son olaraq, bu məqalənin əsas məqsədi informasiya sistemlərinin generasiyasının daha da optimallaşdırılması üçün MGN arxitekturasında model hissəsi üçün yeni yanaşmanın təqdim edilməsi idi. XMA modeli artıq bir real layihədə təqbiq edilib və bizim gələcək layihələrimizdə də tətbiq edilməsi planlaşdırılır.

XMA modelinin tətbiqi və dəyərləndirilməsi bizi hələ ki, ən yaxşı yanaşma olduğuna inandırır.

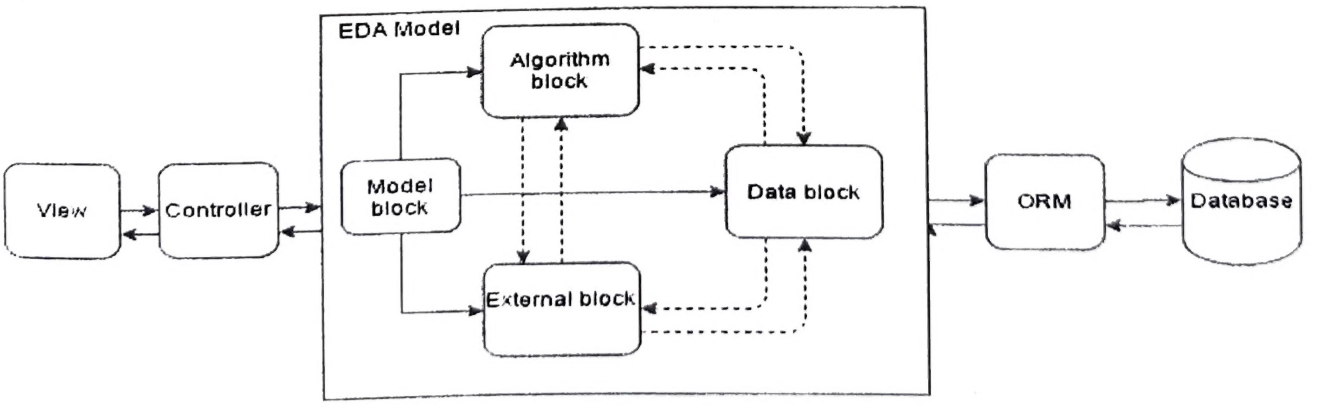

Figure 1: XMA modeli

Son olaraq artıq vurğulandığı kimi, XMA modelinin **xarici servis bloku** da var. Daha öncəki paraqraflarda biz dəqiq nəticənin əldə edilməsi üçün,

ƏDƏBİYYAT

1. **Məqalə:** PK.Ragunath, S.Velmourougan, P. Davachelvan, S.Kayalvizhi, R.Ravimohan , Evolving A New Model (SDLC Model-2010) For Software Development Life Cycle (SDLC), IJCSNS International Journal of Computer Science and Network Security, VOL.10 No.1, January 2010. **2. Məqalə:** Viviana Yarel Rosales-Morales, Giner Alor-Hernández, Jorge Luis García-Alcaráz, Ramón Zatarain-Cabada, María Lucía Barrón-Estrada, An analysis of tools for automatic software development and automatic code generation, Revista Facultad de Ingeniería, Universidad de Antioquia, No. 77, pp. 75-87, 2015. **3. Məqalə:** Nadir Gulzar, "Practical J2EE Application Architecture" Ch.4, March 2003,

Оценка нового подхода (подхода eda - 2017) к генерации кода на основе модели

Н. Ш. Садили, Д. Н. Исмаилзаде

Одной из последних проблем в разработке программного обеспечения является автоматизация процесса разработки программного обеспечения. Существуют различные попытки, чтобы эффективно автоматизировать разработку программного обеспечения, сократить человеческий труд и ускорить его. В этой статье представлена одна из актуальных проблем, связанных с автоматизацией процесса разработки программного обеспечения – генерация кода с использованием модели и объяснение нового подхода EDA. Это описательная статья, и метод качественного анализа используется на протяжении всей работы.

Ключевые слова: разработка программного обеспечения, жизненный цикл разработки программного обеспечения, архитектура контроллера-вида-модели.

Evaluation of new approach (eda approach - 2017) to the model-driven code generation

N. Sh. Sadili, J. N. Ismayilzada

One of the recent challenges in software engineering is automation of software development process. There are various endeavors to efficiently automate software development, reduce human labor and speed it up. This article introduces one of the problems regarding to automation of software development process – model driven code generation and provides explanations of new EDA approach. This is a descriptive article and qualitative analysis method is used throughout the work.

Key words: Software engineering, software development life cycle, model-view-controller architecture